IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Martin Warwick BEALE, et al.

Title: IMPROVEMENTS RELATING TO NETWORKING

Appl. No.: Unassigned

Filing Date: Concurrently herewith

Examiner: Unassigned

Art Unit: Unassigned

PRELIMINARY AMENDMENT

Commissioner for Patents
Washington, D.C. 20231

Sir:

Prior to examination of the present application, Applicant respectfully requests that the above-identified prior application be amended as follows:

In the Claims:

In accordance with 37 C.F.R. § 1.21, please substitute for original Claims 1-25, the following rewritten version of the same claims, as amended. The changes are shown explicitly in the attached "Version With Markings to Show Changes Made."

1. (Amended) A network adapter capable of receiving data from a network, said adapter arranged to receive said data at at least a lowest and a highest data rate; said adapter comprising:

sampling means arranged to sample said data and produce data samples;

an equalizer arranged to receive, and equalize, said data samples, and said equalizer capable of being trained to equalize said data, at at least, each of said lowest and highest data rates; and

training means capable of training said equalizer to equalize said data;

wherein;

said training means is initially arranged to train said equalizer to receive said data at said lowest data rate allowing data to be decoded and if upon decoding said adapter determines that said equalizer has been trained to equalize data at an incorrect rate to retrain said equalizer to equalize data at a correct rate.

2. (Amended) An adapter according to claim 1 that is arranged to receive one or more data sequences and is further arranged such that said equalizer is trained for each said data sequence received.

3. (Amended) An adapter according to claim 1 including a data buffer arranged to receive said data samples from said sampling means.

4. (Amended) An adapter according to claim 3 wherein said data buffer is arranged to store data at said highest data frequency.

5. (Amended) An adapter according to claim 1 wherein said adapter also comprises a training sequence store buffer arranged to receive and store a training sequence held within said data.

6. (Amended) An adapter according to claim 5 wherein said training sequence store buffer is arranged to hold said training sequence at least until it has been determined that said equalizer has been trained to receive said data at said correct rate.

7. (Amended) An adapter according to claim 5 wherein said training sequence store buffer is arranged to receive data from said sampling means.

8. (Amended) An adapter according to claim 3 which also comprises a data down-sampling means arranged to output a selection of said data held in said data buffer to said equalizer.

9. (Amended) An adapter according to claim 5 which also comprises a data down-sampling means arranged to output a selection of said data held in said training sequence store buffer.

10. (Amended) An adapter according to claim 8 wherein said down-sampling means is arranged to output a selection of said data therein by predetermining a constant n, and outputting every n^{th} data sample from said data buffer.

11. (Amended) A method of training an equalizer to equalize a data sequence, which said data sequence may be at one of at least a highest and a lowest data rate, said method comprising:

obtaining at least one data sample by sampling said data sequence;

training said equalizer to receive said data samples at said lowest data rate to provide a trained equalizer;

decoding a portion of said data sequence with said trained equalizer to ascertain the correct data rate; and

retraining said equalizer if said equalizer has been incorrectly trained.

12. (Amended) A method according to claim 11 wherein said method comprises using training data within a preamble of said data sequence to train said equalizer.

13. (Amended) A method according to claim 11 wherein said received data sequence is buffered.

14. (Amended) A method according to claim 13 wherein said data is buffered at said highest data rate.

15. (Amended) A method according to claim 12 wherein training data is held in a training sequence store buffer.

16. (Amended) A method according to claim 11 wherein said method comprises sampling said data sequence at said highest data rate and subsequently using only a portion of the samples should said data sequence be at another data rate.

17. (Amended) A method according to claim 16 wherein said method determines a number n and uses only every n^{th} data sample, in order to use only a

portion of said samples.

18. (Amended) A method according to claim 11 wherein the highest and lowest data rates are part of the series x^y, where x and y are integers.

19. (Amended) A method according to claim 18 wherein x is two.

20. (Amended) A method according to claim 11 wherein said equalizer is trained for each data sequence received.

21. (Amended) A method according to claim 11 wherein said method detects an end of sequence marker within said data sequence and once this is detected returns to a state of waiting to train said equalizer.

22. (Amended) A method according to claim 13 wherein said data sequence comprises a preamble, the end of which is received before buffering of said data samples is enabled.

23. (Amended) A computer readable medium having stored therein instructions for causing a processing unit to execute the method of claim 11.

24. (Amended) A computer program arranged to cause a data sequence to be received, which data sequence may be at one of at least a highest and a lowest data rate, said program:

obtaining samples of said data sequence;

training an equalizer to equalize said samples to provide a trained equalizer;

decoding a portion of said data sequence using said trained equalizer; and

re-training said equalizer if the equalizer has been trained incorrectly at an incorrect data rate.

25. (Amended) An interface including an adapter according to claim 1.

Please add the following new claims 26-35:

26. (New) An interface according to claim 25 arranged to interface a computer or computer peripheral to a network.

27. (New) An interface according to claim 25 provided as any of the following: a PCI card, an ISA card, a USB peripheral, a Firewire peripheral, a PCMIA card, a MODEM riser card.

28. (New) A network adapter capable of receiving data from a network, said adapter arranged to receive said data at at least a lowest and a highest data rate; said adapter comprising:

a sampler arranged to sample said data to produce at least one data sample;

an equalizer arranged to receive, and equalize, said at least one data sample, and said equalizer capable of being trained to equalize said data, at at least, each of said lowest and highest data rates; and

a trainer capable of training said equalizer to equalize said at least one data sample; wherein

said trainer is arranged to initially train said equalizer to receive said data at said lowest data rate allowing said data to be decoded and if upon decoding said data said adapter determines that said equalizer has been trained to equalize said data at an incorrect rate to retain said equalizer to equalize said data at a correct rate.

29. (New) A network adapter capable of receiving data from a network, said adapter arranged to receive a plurality of data sequences each said data sequence containing said data, said adapter arranged to receive said data at at least a lowest and a highest data rate; said adapter comprising:

a sampler arranged to sample said data to produce at least one data sample;

an equalizer arranged to receive and equalize said at least one data sample, and said equalizer being capable of being trained to equalize said data at at least, each of said lowest and highest data rates and said equalizer being arranged to be trained for each data sequence received;

and a trainer capable of training said equalizer to equalize said at least one data sample wherein said trainer is arranged to initially train said equalizer to receive said data at said lowest data rate allowing a portion of said data to be decoded and if, upon decoding said portion of said data, said adapter determines that said equalizer has been trained to equalize said data at an incorrect rate said trainer is arranged to retrain said equalizer to equalize said data at a correct rate.

30. (New) An adapter according to claim 29 which comprises a training sequence store buffer arranged to receive an store a training sequence held within said data.

31. (New) An adapter according to claim 30 wherein said training sequence store buffer is arranged to hold said training sequence at least until it has been determined that said equalizer has been trained to receive said data at said correct rate.

32. (New) An adapter according to claim 31 wherein a data buffer is provided and arranged to buffer said data until is determined that said equalizer has been trained to receive said data at said correct rate.

33. (New) An adapter according to claim 29 wherein said portion of said data comprises a header containing said correct data rate at which said equalizer should be trained.

34. (New) A method of training an equalizer to equalize a plurality of data sequences, which said data sequences may be at one of at least a highest and a lowest data rate, said method comprising:

obtaining at least one data sample by sampling said data sequence;

training said equalizer to receive said at least one data sample at said lowest data rate to provide a trained equalizer;

decoding a portion of said data sequence with said trained equalizer to ascertain the correct data rate;

retraining said equalizer if said equalizer has been incorrectly trained; and

restarting the training process once a data sequence has been completely received to allow the next data sequence to be received.

35. (New) A method according to claim 34 wherein said portion of said data sequence comprises a header containing said data rate at which said equalizer should be trained.

REMARKS

Applicant respectfully requests that the foregoing amendments be made prior to examination of the present application.

Respectfully submitted,

Date January 7, 2002

By 34371

William T. Ellis
Attorney for Applicant
Registration No. 26,874

FOLEY & LARDNER
Washington Harbour
3000 K Street, N.W., Suite 500
Washington, D.C. 20007-5109
Telephone: (202) 672-5485
Facsimile: (202) 672-5399

10/030041

Atty. Dkt. No. 30002458-5

VERSION WITH MARKINGS TO SHOW CHANGES MADE

Marked up Rewritten Claims:

1. (Amended) A network adapter capable of receiving data from a network, said adapter arranged to receive said data at at least a lowest and a highest data rate; said adapter comprising:

sampling means arranged to sample said data;

an equalizer [equaliser] arranged to receive, and equalize [equalise], said data samples, and said equalizer [equaliser] capable of being trained to equalize [equalise] data, at at least, each of said [lower] lowest and [higher] highest data rates; and

training means capable of training said equalizer [equaliser] to equalize [equalise] data;

wherein;

said training means is initially arranged to train said equalizer [equaliser] to receive data at said [lower] lowest rate allowing data to be decoded and if upon decoding said adapter determines that said equalizer [equaliser] has been trained to equalize [equalise] data at an [the] incorrect rate to retrain said equalizer [equaliser] to equalize [equalise] data at [the] a correct rate.

2. (Amended) An adapter according to claim 1 that is arranged to receive one or more data sequences and is further arranged such that said equalizer is trained for each said data sequence received [including a data buffer arranged to receive sampled data from said sampling means].

3. (Amended) An adapter according to claim 1 including a data buffer arranged to [2 wherein the data buffer is arranged to store data at the highest frequency] receive said data samples from said sampling means.

4. (Amended) An adapter according to claim 3 [any one of the preceding claims] wherein said data buffer is arranged to store data at said highest data frequency [the adapter comprises a second buffer, or a training sequence store buffer,

arranged to receive and store training data held within the received data].

5. (Amended) An adapter according to claim 1 [4] wherein [the training sequence store buffer is arranged to hold the training sequence at least until it has been determined that the equaliser has been trained to receive data at the correct rate] said adapter also comprises a training sequence store buffer arranged to receive and store a training sequence held within said data.

6. (Amended) An adapter according to claim [4 or] 5 wherein [the] said training sequence store buffer is arranged to hold said training sequence at least until it has been determined that said equalizer has been trained to receive said data at said correct rate [the training sequence store buffer receives data from the sampling means].

7. (Amended) An adapter according to claim 5 wherein said training sequence store buffer is arranged to receive data from said sampling means [any one of claims 2 to 6 wherein the adapter comprises data down-sampling means arranged to output a selection of the data held in a buffer to the equaliser].

8. (Amended) An adapter according to claim [7] 3 which also comprises a data down-sampling means arranged to output a selection of said data held in said data buffer to said equaliser [wherein the down-sampling means is arranged to output a selection of the data therein, by predetermining a constant n, and outputting every nth data sample from the buffer].

9. (Amended) An adapter according to claim 5 which also comprises a data down-sampling means arranged to output a selection of said data held in said training sequence store buffer [A method of training an equaliser to equalise a data sequence, which data sequence may be at one of at least a highest and a lowest data rate, said method comprising;

obtaining a number of data samples by sampling said data sequence;

training said equaliser to receive said data sample at said lowest data rate;

decoding a portion of said data sequence with the trained equaliser to ascertain the correct data rate; and

retraining said equaliser if the equaliser has been incorrectly trained].

10. (Amended) [A method according to claim 9 wherein the method comprises using training data within a preamble of the data sequence to train the equaliser] An adapter according to claim 8 wherein said down-sampling means is arranged to output a selection of said data therein by predetermining a constant n, and outputting every nth data sample from said data buffer.

11. (Amended) [A method according to claim 9 or 10 wherein the received data sequence is buffered.] A method of training an equalizer to equalize a data sequence, which said data sequence may be at one of at least a highest and a lowest data rate, said method comprising:

obtaining at least one data sample by sampling said data sequence;

training said equalizer to receive said data samples at said lowest data rate to provide a trained equalizer;

decoding a portion of said data sequence with said trained equalizer to ascertain the correct data rate; and

retraining said equalizer if said equalizer has been incorrectly trained.

12. (Amended) A method according to claim 11 wherein said method comprises using training data within a preamble of said data sequence to train said equalizer [data is held in the buffer at the highest data rate].

13. (Amended) A method according to claim 11 wherein said received data sequence is buffered [any one of claims 10 to 12 wherein training data is held in a second, training sequence store, buffer].

14. (Amended) A method according to claim 13 [any one of claims 9 to 13] wherein said data is buffered at said highest [the method comprises sampling the data sequence at the highest data rate and subsequently using only a portion of the samples should the data sequence be at another] data rate.

15. (Amended) A method according to claim 12 [14] wherein training data is held in a training sequence store buffer [the method uses only a portion of the data samples by determining a number n and using only every nth data sample].

16. (Amended) A method according to claim 11 [any one of claims 9 to 15] wherein said method comprises sampling said data sequence at said highest data rate and subsequently using only a portion of the samples should said data sequence be at another data rate [the highest and lowest data rates are part of the series x^y, where x and y are integers].

17. (Amended) A method according to claim 16 wherein said method determines a number n and uses only every nth data sample, in order to use only a portion of said samples [x is two].

18. (Amended) A method according to claim 11 wherein the highest and lowest data rates are part of the series x^y, where x and y are integers [any one of claims 9 to 17 wherein the equaliser is trained for each data sequence received].

19. (Amended) A method according to any one of claim [claims 9 to] 18 wherein x is 2 [the method detects an end of sequence marker within the data sequence and once this is detected returns to a state of waiting to train the equaliser].

20. (Amended) A method according to claim 11 wherein said equalizer is trained for each data sequence received [any one of claims 9 to 19 wherein the data sequence comprises a preamble, the end of which is received before inputs to the, or each, buffer are enabled for received data samples].

21. (Amended) A method according to claim 11 wherein said method detects an end of sequence marker within said data sequence and once this is detected returns to a state of waiting to train said equalizer [A computer readable medium having stored therein instructions for causing a processing unit to execute the method of any of claims 9 to 20].

22. (Amended) A method according to claim 13 wherein said data sequence comprises a preamble, the end of which is received before buffering of said data samples is enabled [A computer program arranged to cause a data sequence to be received, which data sequence may be at one of at least a highest and a lowest data rate, said program:

obtaining samples of said data sequence;

training an equaliser to equalize said samples;

decoding a portion of said data sequence using said trained equaliser; and

re-training said equaliser if the equaliser has been trained incorrectly to the incorrect data rate].

23. (Amended) A computer readable medium having stored therein instructions for causing a processing unit to execute the method of claim 11 [An interface including an adapter according to any one of claims 1 to 8].

24. (Amended) A computer program arranged to cause a data sequence to be received, which data sequence may be at one of at least a highest and a lowest data rate, said program: [An interface according to claim 23 arranged to interface a computer or computer peripheral to a network.]

obtaining samples of said data sequence;

training an equalizer to equalize said samples to provide a trained equalizer;

decoding a portion of said data sequence using said trained equalizer; and

re-training said equalizer if the equalizer has been trained incorrectly at an incorrect data rate.

25. (Amended) An interface including an adapter according to claim 1 [23 or 24 provided as any of the following: a PCI card, an ISA card, a USB peripheral, a Firewire peripheral, a PCMIA card, a MODEM riser card].